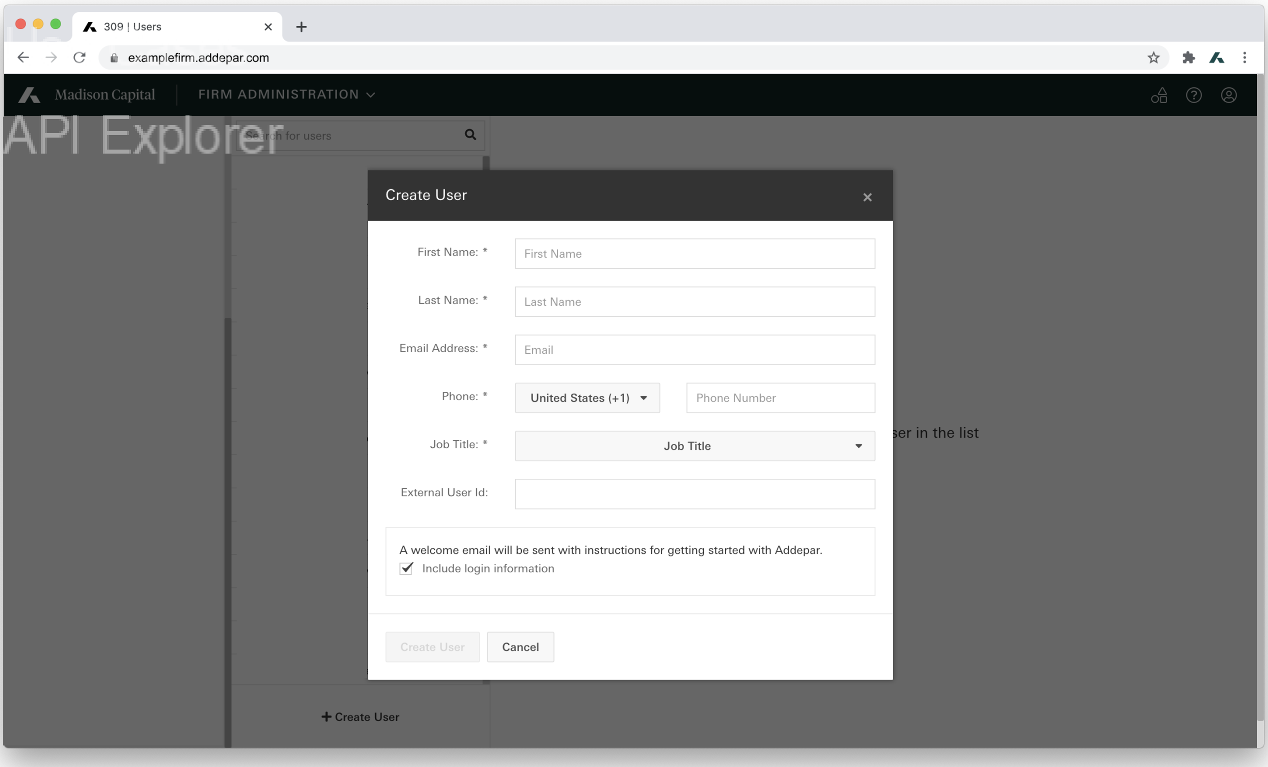
Users

Users are members of your firm that have access to Addepar. The Users API can be used to view, create, and delete users. You can also update user information, tool permissions, and portfolio access.



Create User

First Name: * First Name

Last Name: * Last Name

Email Address: * Email

Phone: * United States (+1) Phone Number

Job Title: * Job Title

External User Id:

A welcome email will be sent with instructions for getting started with Addepar.

☑ Include login information

Create User Cancel

Base Route	/v1/users
Endpoints	**GET** /v1/users/:id /v1/users /v1/users/:id/relationships/assigned_role /v1/users/:id/relationships/permissioned_entities /v1/users/:id/relationships/permissioned_groups **POST** v1/users/email_query /v1/users/external_user_id_query /v1/users /v1/users/:id/relationships/permissioned_entities /v1/users/:id/relationships/permissioned_groups **PATCH** /v1/users/:id /v1/users/:id/relationships/assigned_role **DELETE** /v1/users/:id
Produces	JSON
Pagination	Yes
Application Permissions Required	"API Access: Create, edit, and delete" "Manage firm settings: Users and permissions" for all operations.

OAuth Scopes	**GET**
	`USERS_WRITE`
	POST, **PATCH**, and **DELETE**
	`USERS_WRITE` except **POST** `v1/users/external_user_id_query` which requires `USERS` or `USERS_WRITE`

Resource Overview

Users are described by the below resource object attributes. Attributes required for creating, updating, or deleting Users are noted.

All attributes will be returned in successful **GET**, **POST** & **PATCH** responses containing the Users resource.

ATTRIBUTE	DESCRIPTION	EXAMPLE
`email`	The email address used for authentication. Not editable. String.	`"user1@addepar.com"`
`first_name`	String.	`"Adam"`
`last_name`	String.	`"Smith"`
`login_method`	Not editable. String. Supported values: - email_password - saml	`"email_password"`
`saml_user_id`	Not editable. String. Required if login_method is "saml".	`"asmith"`
`admin_access`	Indicates whether the user has access to all permissions. Boolean.	`false`
`all_data_access`	Indicates whether the user has permission to access all current and future portfolio data. Boolean.	`true`
`two_factor_auth_enabled`	Indicates whether the user has two-factor authentication enabled. Not editable. Boolean.	`true`
`external_user_id`	Reflects the firm’s unique ID for the user, like the employee's ID number or ID from a human resources system, etc. String.	`"A67890"`

Relationships

RELATIONSHIP	DESCRIPTION
`assigned_role`	The role that the user is assigned to.
`permissioned_entities`	The client portfolios the user has access to.
`permissioned_groups`	The groups the user has access to.

RELATIONSHIPS

```
"relationships": {
        "permissioned_entities": {
          "links": {
            "self": "/v1/users/2000/relationships/permissioned_entities",
            "related": "/v1/users/2000/permissioned_entities"
          },
          "data": [
            {
              "type": "entities",
              "id": 10000
            },
            {
              "type": "entities",
              "id": 10001
            }
          ]
        },
        "assigned_role": {
          "links": {
            "self": "/v1/users/2000/relationships/assigned_role",
            "related": "/v1/users/2000/assigned_role"
          },
          "data": []
        },
        "permissioned_groups": {
          "links": {
            "self": "/v1/users/2000/relationships/permissioned_groups",
            "related": "/v1/users/2000/permissioned_groups"
          },
          "data": [
            {
              "type": "entities",
              "id": 20000
            },
            {
              "type": "entities",
              "id": 20001
            }
          ]
        }
      }
```

Get a User

Retrieves details for a specific user.

GET `/v1/users/:id`

Example:

REQUEST RESPONSE

```
GET https://examplefirm.addepar.com/api/v1/users/2000
```

Responses

- `200 OK` : Success

- `403 Forbidden` : User lacks sufficient application permissions
- `404 Not Found` : Nonexistent/non-permissioned user ID

Get All Users

Retrieves details for all users.

GET `/v1/users`

Example:

REQUEST RESPONSE

```
GET https://examplefirm.addepar.com/api/v1/users
```

Responses:

- `200 OK` : Success
- `403 Forbidden` : Lacking permission to view users

Get a User's Assigned Role

If no role is assigned to a user, data will return as `null` .

GET `/v1/users/:user-id/relationships/assigned_role`

Example:

REQUEST RESPONSE

```
GET https://examplefirm.addepar.com/api/v1/users/101/relationships/assigned_role
```

Response Codes:

- `200 OK` : Success
- `400 Bad Request` : Invalid relationship queried
- `403 Forbidden` : User lacks sufficient application permissions
- `404 Not Found` : Nonexistent/non-permissioned user ID

Get a User's Permissioned Entities Or Groups

Retrieves a list of IDs for the clients or group portfolios that the user has access to.

GET `/v1/users/:user-id/relationships/permissioned_entities`

GET `/v1/users/:user-id/relationships/permissioned_groups`

Example:

REQUEST RESPONSE

```
GET https://examplefirm.addepar.com/api/v1/users/5678/relationships/permissioned_groups
```

Response Codes:

- `200 OK` : Success
- `400 Bad Request` : Invalid relationship queried
- `403 Forbidden` : User lacks sufficient application permissions
- `404 Not Found` : Nonexistent/non-permissioned user ID

Get Users by Email

Retrieves details for a specific user.

POST `/v1/users/email_query`

Example:

REQUEST RESPONSE

```
POST https://examplefirm.addepar.com/api/v1/users/email_query

{
  "data":{
    "type":"email_query",
    "attributes":{
      "email_ids":[
        "user1@addepar.com",
        "user2@addepar.com"
      ]
    }
  }
}
```

Response Codes:

- `200 OK` : Success
- `403 Forbidden` : User lacks sufficient application permissions

Get Users by External User ID

Retrieves details for a specific user.

POST `/v1/users/external_user_id_query`

Example:

REQUEST RESPONSE

```
POST https://examplefirm.addepar.com/api/v1/users/external_user_id_query

{
  "data":{
    "type":"external_user_id_query",
    "attributes":{
      "external_user_ids":[
        "A12345",
        "A67890"
      ]
    }
  }
}
```

Responses:

- `200 OK` : Success
- `403 Forbidden` : User lacks sufficient application permissions

Create a User

By default, users are created in custom mode with no permissions. You can keep them in custom mode and manually assign permissions to each user in the Addepar application.

If you want to assign user permissions based upon a specific role, you can do so using the Update User Relationships method in the Users API or the Assign Role to Users method in the Roles API.

POST `/v1/users`

Example:

REQUEST RESPONSE

```
POST https://examplefirm.addepar.com/api/v1/users

{
  "data":{
    "type":"users",
    "attributes":{
      "email":"example.user@addepar.com",
      "first_name":"Example",
      "last_name":"User",
      "login_method":"email_password"
    }
  }
}
```

Response Codes:

- `201 Created` : Success
- `400 Bad Request` : Invalid email provided
- `400 Bad Request` : SAML User ID already in use
- `400 Bad Request` : Email is already in use
- `403 Forbidden` : User lacks sufficient application permissions

- 409 Conflict : A duplicate external_user_id exists for the firm

Add User's Access to Entities or Groups

Grants a user access to a specific client or group of portfolios.

POST /v1/users/:id/relationships/permissioned_entities

POST /v1/users/:id/relationships/permissioned_groups

Example:

REQUEST RESPONSE

```
POST https://examplefirm.addepar.com/api/v1/users/101/relationships/permissioned_groups

{
  "data":{
    "id":"10",
    "type":"groups"
  }
}
```

- 204 No Content : Success
- 400 Bad Request : Nonexistent/non-permissioned client IDs
- 404 Not Found : Nonexistent/non-permissioned user ID

Update a User

Updates the user's first_name , last_name , all_data_access , or admin_access .

PATCH /v1/users/:id

Example:

REQUEST RESPONSE

```
PATCH https://examplefirm.addepar.com/api/v1/users/101

{
  "data":{
    "type":"users",
    "id":"621500",
    "attributes":{
      "first_name":"Second",
      "last_name":"User"
    }
  }
}
```

Response Codes

- 200 OK : Success
- 400 Bad Request : Attempted to update relationships

- `403 Forbidden` : User lacks sufficient application permissions
- `404 Not Found` : Nonexistent/non-permissioned user ID
- `409 Conflict` : A duplicate external user ID exists for the firm

Update a User's Role

> **Note**
>
> Before using this method, you must assign a role to a user in the Addepar application.

PATCH `/v1/users/:id/relationships/assigned_role`

Example:

REQUEST RESPONSE

```
PATCH https://examplefirm.addepar.com/api/v1/users/101/relationships/assigned_role

{
  "data":{
    "id":"1",
    "type":"role"
  }
}
```

Responses

- `204 No Content` : Success
- `400 Bad Request` : Nonexistent/non-permissioned role ID
- `404 Not Found` : Nonexistent/non-permissioned user ID

Delete a User

Removes a specified user.

DELETE `/v1/users/:id`

Example:

REQUEST RESPONSE

```
DELETE https://examplefirm.addepar.com/api/v1/users/101

{
  "data":{
    "type":"users",
    "id":"users",
    "attributes":{
      "first_name":"Example",
      "last_name":"User"
    }
  }
}
```

Response Codes:

- `204 No Content` : Success
- `403 Forbidden` : User lacks sufficient application permissions
- `404 Not Found` : Nonexistent/non-permissioned user ID

Delete a User's Access to Entities or Groups

Removes access to specific client and group portfolios.

DELETE `/v1/users/:id/relationships/permissioned_entities`

DELETE `/v1/users/:id/relationships/permissioned_groups`

Example:

REQUEST RESPONSE

```
DELETE https://examplefirm.addepar.com/api/v1/users/101/relationships/permissioned_groups

{
  "data":[
    {
      "id":"1",
      "type":"groups"
    }
  ]
}
```

Response Codes:

- `400 Bad Request` : Invalid relationship queried
- `403 Forbidden` : User lacks sufficient application permissions
- `404 Not Found` : Nonexistent/non-permissioned user ID

Jobs API

Portfolio API

Positions API

Roles API

Transactions API

Users API

Integrations

Contact Us

Partner with Addepar

Privacy Policy | Terms of Use